April 1975

INWG PROTOCOL NOTE #21
by G.Le Moli

A PROPOSAL FOR FRAGMENTING PACKETS IN INTERNETWORKING

This note proposes an internetwork fragmentation mechanism which could be used in an ent to end (host-to-host) internetwork protocol hereinafter referred as EE-protocol: such mechanism allows an high degree of freedom in establishing the rules of the EE-protocol using this mechanism.

The CK protocol [1] has the advantage that allows any kind of fragmentation having the octet as indivisible unit: this cost a complicated method for acknowledgment (window mechanism); on the contrary, ZE protocol, [2], has a rigid length of the fragment which avoid such complications, but either forces to choose as international packet length the minimum one among the interconnected networks, or forces a network having a packet length shorter than the choosen international length to fragment and then to rebuild an international packet in transit. ZE protocol however uses a window mechanism having as unit the letter, that is a group of 127 packets (the unit which is acknowledged in YR-REF is infact the letter).

McKenzie [3] performs a very good comparison between CK and ZE protocols and suggests a compromize in which the main idea is to use a mechanism of the CK type having an unit much higher than the octet and much lower than the letter and proposes a method for computing a good value for such unit. In this way, a substantial improvment of both CK and ZE protocols can be achieved.

The unit which McKenzie suggests is therefore the minimum data field packet length which is supposed to be accepted by all the networks. In this case the couple (MY-REF, FR-NB) identifies a packets and allows to fragment it: however McKenzie does not explicity suggest that the window mechanism which CK applies on the basis of the octet and which ZE applies on the basis of the letter should be applied on the basis of the unit of McKenzie.

The mechanism here proposed can be compared with the mechanism of McKenzie on the following items:

- both use a unit much higher than one octet and much lower than a letter
- let us indicate as L_i the maximum data field which each network allows for its packets; let L_m be the minimum value and L_M be the maximum value of L_i among all the L_i: McKenzie fixes the value of L_i, the method here proposed required that $L_M/L_m \leq 128$, that is the shortest Data Field should not be shorter than 1/128 of the longest Data Field;
- the mechanism here proposed does not require to know anything about the used EE-protocol, except the length of the EE-header and the position of one octet the use of which must be reserved to the mechanism here proposed in the gateways which needs to fragment packets.

Fig.1 reproduces an international packet in which only what is here concerned with is drawn:

- the first region contains local and/or international packet header;
- the second region contains the EE-header, in which one octet is reserved for gateways using the mechanism here proposed: this octet will be hereinafter referred to as the LB octet;
- the third region contains the EE-text.

Somewhere, there exists a packet identifier and/or a packet sequencing number; it is irrilevant wether this information is in the first or in the second region or in both: surely the receiver known where it is.
The method here proposed is only concerned with LB octet in the EE-header region: such octet is set to X'01' by the sender and is not changed until some gateway needs to fragment the packet. In this case it produces two packet and inserts in the LB octets of such new packets suitable information which will allow the receiver to rebuild the original packet.
The leading idea is that each time one packet is fragmented, it is fragmented into two (or 2^k) packets: the two new packets are considered its "children" and the LB octets carry information regarding the "relationships" between the members or the family of packets generated by successive fragmentations.

The original packets will be called "a first generation packet"; their children will be the second generation and so on. In Fig. 2 the LB octets of the first, second,..., fifth generation are shown. In Table 1 the rules for assigning bit patterns to LB octets are shown for all the possible eigth generations.Generally, LB octets of packets of the n-th generation have the following bit patterns: the n-th bits from the right is set to 1, the n-1 bit on the right give a number from 0 to $2^{n-1}-1$ as label for the 2^{n-1} members of the n-th generation; the 8-n bits on the left are set to 0.
It is easy to recognize that the following rule is valid: when a packet is split into two children, their LB octets are obtained shifting one bit to the left the LB octet of the father, with the new bit on the right set to to 0 for the first child and to 1 for the second one.
It is convenient that each time a fragmentation occours, the fragmentation produces 2^k fragments of nearly equal lengths, even if a lower number of fragments could be enough: infact this allows to reach the ratio $L_M/L_m = 128$ (it can be shown that with some value of L_i this ratio is nor reached: e.g. with L_i slightly decreasing) and semplifies considerably the procedure for reassembling the original packet at the receiver side.
Let us now suppose that the packet of fig.1 arrives at a gateway which must divide this packet into two children: both the children packets contain the same first region as the father packet did, except for the Data Field lengths; both contain the same EE-header as the father, except the LB octet; both contain half of the EE-text of the father.
The sender is only required to set to X'01' the LB octet in each of its packets; the receiver needs to test the LB field: if it is X'01',then the packet is forwarded to the EE-protocol, else it is given to a specialized Assembling Routine (AR) which waits for the arrival of all the members of that family, and then rebuild the original packet which is finally delivered to the EE-protocol, which does not even need to be aware of the fact that a particular packet has been fragmented or not; note that AR can discharge duplicate members of the family, and must set up a reassembly time-out:

in this way a lost member of the family causes the EE-protocol to believe that the whole original packet has been lost (1). ACK's and NACK's are sent by EE-protocol on the basis of the original packets.

AR can distinguish different families of packets because all the members of one family carry the same Identifier Field and the same sequencing number as the original packet did.

The main characteristics of the mechanism here proposed can be summarized as follows:

- it needs only one octet, the LB octet;
- it does require to know anything neither about the header of the packet nor about the EE-protocol which is used, except the position of the LB octet and possibly the length of EE-header (if the "Message Identifier" of the packet contains suitable information, the knowledge of the length of the EE-header is not required any more);
- the EE-protocol is not made aware of the fact that a packet has been frag mented or not: therefore the EE-protocol can be established without taking into account the fragmentation mechanism;
- each receiver can implement its AR indipendently from the other receivers and accordling to the best use of the hardware and software resources which are there available

---

(1) The EE-protocol could also be such that the reassembling routine can ask the sender to repeat only some fragments of the original packet.

REFERENCES

[1] V.Cerf, R.Kahn :"Towards protocol for Internetwork Communication" If INWG 39'

[2] H. Zimmermann, M. Elie :"Transport protocol, Standard Protocol for Heterogeneus Network" INWG 43

[3] A. Mc Kenzie :"Internetwork Host-to-Host Protocol" INWG 74

PACKET GENERATION	BIT PATTERN FOR MEMBERS OF THIS GENERATION
first	00000001
second	0000001 ; bit 8 labels the two members of this generation
third	000001 ; bits 7 and 8 label from 0 to 3 the 4 members
fourth	00001 ; bits from 6 to 8 label from 0 to 7 the 8 members
fifth	0001 ; " " 5 " 8 " " 0 " 15 " 16 "
sixth	001 ; " " 4 " 8 " " 0 " 31 " 32 "
seventh	01 ; " " 3 " 8 " " 0 " 63 " 64 "
eighth	1 ; " " 2 " 8 " " 0 " 127 " 128 "

TABLE 1

Local and/or international header: contains the Data Field length	EE-header: contains the LB octec	EE-Text

DATA FIELD

Fig. 1

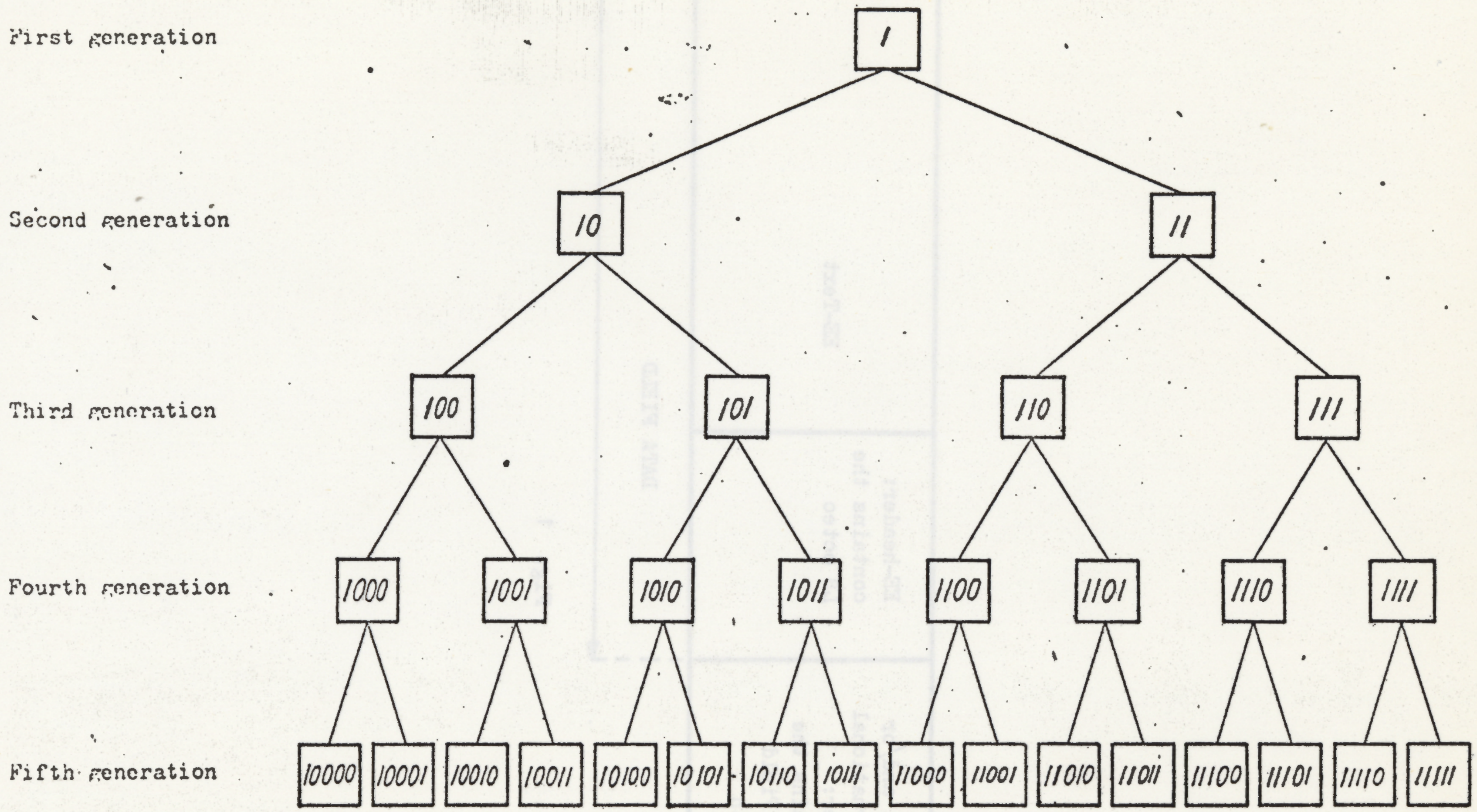
First generation
Second generation
Third generation
Fourth generation
Fifth generation
1
10
11
100
101
110
111
1000
1001
1010
1011
1100
1101
1110
1111
10000
10001
10010
10011
10100
10101
10110
10111
11000
11001
11010
11011
11100
11101
11110
11111

Fig 2